# Y

REDACTED VERSION

IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

SRI INTERNATIONAL, INC.,
a California Corporation,

Plaintiff and
Counterclaim-Defendant,

v.

INTERNET SECURITY SYSTEMS, INC.,
a Delaware Corporation, INTERNET
SECURITY SYSTEMS, INC., a Georgia
Corporation, and SYMANTEC
CORPORATION, a Delaware Corporation,

Defendants and
Counterclaim-Plaintiffs.

C. A. No.: 04-1199 (SLR)

**DECLARATION OF L. TODD HEBERLEIN**

## TABLE OF CONTENTS

I, L. Todd Heberlein, declare that:

1. I am the President of Net Squared, Inc.

2. I have been retained by counsel for Symantec Corporation as an expert witness in this action. If called to testify as to the truth of the matters stated herein, I could and would do so competently.

3. I received a Bachelor of Science degree in Computer Science and Math from the University of California, Davis in 1988, and a Masters of Science degree in Computer Science from the University of California, Davis in 1991.

4. I was the primary developer of the first network-based intrusion detection system, the Network Security Monitor (NSM), at UC Davis in the late 1980s and early 1990s. The NSM processed network packets and applied anomaly and signature detection techniques to detect intrusive activity. The work began in 1988, and we published numerous papers describing the work in 1990, 1991, and 1994. By the mid-1990s the NSM was deployed at numerous organizations including UC Davis, the Air Force, the Department of Energy, NASA, and the Department of Justice. The Air Force deployed it at over 100 Air Force sites globally. Within the Air Force and at Lawrence Livermore National Laboratory, the NSM was usually deployed at the organization's gateway to the rest of the Internet.

5. I was also one of the primary developers for the first hierarchical and distributed intrusion detection system, the Distributed Intrusion Detection System, (DIDS) which integrated NSM, host monitors for SunOS and VMS, and a centralized Director. DIDS was deployed at UC Davis and Lawrence Livermore National Laboratory in 1992. I have been told that the Air Force deployed DIDS at other locations at a later date.

6. From 1993 to 1995 I worked on a DARPA contract called Intrusion Detection for Large Networks, and in 1995 I presented results, including an initial fully distributed intrusion detection capability, to our DARPA Program Manager, Teresa Lunt.

That work eventually became the Graph-based Intrusion Detection System (GrIDS). In addition to intrusion detection technology, I researched issues about the fundamental nature of computer vulnerabilities, and that work won a "best paper" award at the National Information Systems Security Conference in 1996.[1] The paper was also republished in a book by Dorothy and Peter Denning.[2]

7. I founded Net Squared, Inc. in 1996. At Net Squared I performed research and development in computer security for numerous organizations including the Air Force, Lawrence Livermore National Laboratory, the Defense Advanced Research Projects Agency (DARPA), Office of Naval Research, the Intelligence Community, and the Federal Aviation Administration. At Net Squared I developed the Network Monitoring Framework (NMF), a library of network monitoring C++ objects, and Network Radar, a suite of network monitoring applications built on the NMF libraries. Network Radar tools were integrated into larger, hierarchical intrusion detection systems by the Air Force and Boeing, including EPIC, EPIC2, AIDE, AFED, and IDIP. Network Radar was deployed at UC Davis, the Air Force's Rome Research Laboratory, and during numerous Air Force exercises. From approximately 2002-2003 I also led a project called TrendCenter, which integrated and correlated intrusion detection alerts from unrelated organizations. As part of that effort I developed SANS' initial Internet Storm Center prototype.

8. In addition to traditional intrusion detection capabilities (anomaly detection and signature detection), I developed or help develop several other intrusion detection technologies. For example, we profiled network services in NSM and DIDS,

[1] L. T. Heberlein, M. Bishop, "Attack Class: Address Spoofing," 19th National Information Systems Security Conference, Baltimore, MD, 22-25 Oct. 1996, pp. 371-377.

[2] D. Denning and P. Denning, INTERNET BESIEGED, COUNTERING CYBERSPACE SCOFFLAWS, 1st ed. (Oct. 3, 1997) at Chap. 10.

and that work was also part of a feature selection effort led by Jeremy Frank.[3] The profiling of network services work, later based on a feed-forward back-propagation neural network, was also rolled into Network Radar's Non-Cooperative Service Recognition (NCSR) technology.

9. I also did the initial work on network thumbprinting that was introduced in the 1992 Internetwork Security Monitor paper,[4] and I helped refine the thumbprinting technology that was published in a 1995 IEEE paper.[5] Thumbprinting uses a multivariate statistical technique called principal component analysis to reduce a high dimensional object to a small dimensional object. Vector distances of the lower dimensional objects were then used to determine if the objects were correlated.

10. Both the NCSR and the thumbprinting work were also rolled into Network Radar, which was able to detect the earliest stages of the ILOVEYOU worm as it hit the Air Force's Rome Labs.

11. A summary of my professional experience and publications are attached as Exhibit A.

12. I receive compensation in the amount of $258.00 per hour for the time that I devote to this matter. My compensation is not dependent in any way on the outcome of this matter.

[3] J. Frank , "Machine Learning and Intrusion Detection: Current and Future Directions," Proc. of the 17th National Computer Security Conference, October 1994.

[4] L.T. Heberlein, B. Mukherjee, K.N. Levitt., "Internetwork Security Monitor: An Intrusion-Detection System for Large-Scale Networks," Proc. 15th National Computer Security Conference, pp. 262-271, Oct. 1992.

[5] S. Staniford-Chen, and L.T. Heberlein, "Holding Intruders Accountable on the Internet," Proc. of the 1995 IEEE Symposium on Security and Privacy, Oakland, CA, 8-10 May 1995, pp. 39-49.

## I. METHODOLOGY AND BASES

13. In preparing my opinions and analysis of the art, I have thoroughly reviewed the entire specification and claims of U.S. Patents No. 6,321,338 (the '338 patent); 6,484,203 (the '203 patent); 6,708,212 (the '212 patent); and 6,711,615 (the '615 patent) (collectively, the patents-in-suit). I have also reviewed each of the prosecution histories and the Microfiche Appendix included with the patents-in-suit.

14. I have also reviewed the expert report of Mr. Frederick Avolio, and have indicated in my declaration instances where I have relied upon this report.

15. I have reviewed an extensive body of prior art publications and product documentation. I have also spoken directly with a number of individuals who I understand and believe to be personally knowledgeable with respect to the prior art embodiments discussed below. A list of the prior art publications and documentation I have reviewed and the individuals with whom I have spoken in forming the opinions set forth below is attached as Exhibit B.

16. I have also compared each of the claims of the patents-in-suit with certain prior art publications and embodiments discussed below.

17. In general, my methodology of inquiry with respect to each of the principal prior art publications and embodiments relied upon in my opinions was as follows:

a. With regard to prior art publications, I reviewed each publication in detail.

b. With regard to product embodiments, I reviewed the marketing literature, product documentation and manuals relating to the prior art system to familiarize myself with the features, functions and capabilities of the system.

c. With regard to product embodiments, I typically also engaged in a lengthy conversation with at least one individual who was personally knowledgeable of the facts relating to the development, marketing and history of the prior art embodiment. In this conversation I attempted both to challenge and to confirm my understanding of the facts that I had derived from my prior investigations.

22. The '203, '212 and '615 patents are all continuations of the '338 patent. I understand that this means that all four patents-in-suit are entitled to the filing date of the '338 patent, which is November 9, 1998. I also understand that as continuations, the '203, '212 and '615 patents may not add "new matter" or additional disclosures to the specification of the '338 patent. Although the "References Cited," "Abstract" and "Summary" sections of the four patents-in-suit differ in some respects, the Figures and "Detailed Description" sections are essentially identical and provide a common description of the alleged inventions. Because the most relevant portions of the written description of the four patents-in-suit share a common disclosure, I have cited only to the '338 patent in my description of the alleged inventions. However, this description applies to all four of the patents-in-suit.

## IV. LACK OF NOVELTY – DESCRIPTION OF PRIOR ART SYSTEMS

23. The following sections discuss the features and functionality of certain prior art publications and systems that in my opinion demonstrate that some or all of the asserted claims are invalid as anticipated.

### A. JiNao

24. This section covers the JiNao design as described in the technical report by F. Jou, "Architecture Design of a Scalable Intrusion Detection System for the Emerging Network Infrastructure," dated April 1997 ("*JiNao Report*"). MCNC and North Carolina State University (NCSU) performed this work. The JiNao design provided a scalable architecture to detect and protect against both known and unknown attacks against the network infrastructure.

25. The architecture described, collectively called JiNao, used a combination of anomaly detection, misuse detection, correlation, and aggregation to detect intrusive activity. The lowest level JiNao monitors (described as a "Local JiNao") analyzed packets, while the higher level monitors (described as "Remote Management

Applications") analyzed the results of Local JiNao monitors or other higher level monitors. Communications between monitors was via the standard Simple Network Management Protocol (SNMP) and Management Information Base (MIB) architecture. JiNao also uses the standard SNMP network protocol and MIB architecture to support hierarchical and peer-to-peer communications to provide scalability and detect larger scale attacks. The JiNao architecture was independent of any particular network protocol (the modules use the generic term "Protocol Data Unit" (PDU)), but the developers planned to deliver implementations for analyzing the OSPF routing protocol and the SNMP management protocol.[6] By placing Local JiNao monitors on routers or next to routers, JiNao could respond to attacks by blocking offending packets. Furthermore, the authors planned to track the DARPA joint effort called Common Intrusion Detection Framework (CIDF) to support interoperability and module reusability with other DARPA sponsored projects.[7]

26. Thus, JiNao had the following salient features:

- Processed network packets to look for intrusive behavior and network faults.
- Ran on or next to routers and gateways.
- Designed to be independent of any particular network protocol.
- Initial implementation would analyze the OSPF routing protocol and the SNMP management protocol.
- Analyzed network traffic with both misuse and anomaly detection engines.
- Used NIDES statistical algorithms for anomaly detection.
- Correlated the results of the misuse and anomaly detectors.

[6] *JiNao Report* at 5-6, 14, 15.

[7] *JiNao Report* at 13-14.

- Correlated encrypted and decrypted versions of the same packet to enhance analysis.
- Responded to attacks by alerting higher-level monitors (via SNMP trap messages), modifying analysis, and blocking traffic.
- Modified analysis based on events detected by other sensors and monitors.
- Exchanged information with other monitors using a standardized network protocol (SNMP) and databases (MIB).
- Used the same misuse and anomaly detection approaches at hierarchically higher monitors.
- Supported peer-to-peer and hierarchical monitoring for higher-level monitors.
- Correlated and aggregated/integrated information at regional and global levels.
- Designed to integrate with additional network management or security applications by using the standards-based SNMP framework.
- Designed to support interoperability and reusability with other DARPA-sponsored projects efforts via the emerging Common Intrusion Detection Framework (CIDF).

27. Part of a local JiNao system is the Intercept/Redirect module which intercepted a network packet before the router's software could process it, forwarded the packet to the analysis modules, and would forward the packet to the router's software. If JiNao determined that a packet was part of an attack, this module could respond by preventing the malicious packets from being sent to the router's software.

28. The Intercept/Redirect module forwarded the packet to the next module in the JiNao system, the Simple Misuse Detection module. The *JiNao Report* actually calls this the rule-based "prevention module," and it applied simple rules to a single packet in order to quickly detect clear security violations.

29. The Simple Misuse Detection module forwarded the packets to the local detection module that consisted of an anomaly detector and a misuse detector that

detected attacks that spanned multiple packets. The anomaly and misuse detectors forwarded their results to the Local Decision Module, which correlated information from the other sensors and determined whether a response should be taken. Potential responses included directing the Simple Misuse Detector to start blocking certain packets, adjusting the analysis of the misuse and anomaly detectors, making the analysis available to other monitors through a MIB, and alerting high-level monitors through an SNMP trap message.

30. A local JiNao system communicated with a hierarchically higher monitor through the network agent using a standardized SNMP protocol. JiNao's local results and configuration information were available to higher-level monitors through a Management Information Base (MIB), and alerts were automatically sent to higher-level monitors though SNMP trap messages, part of the standardized SNMP network management architecture.

31. The JiNao architecture supported hierarchically higher monitors called Remote Management Applications that used the same anomaly and misuse algorithms as the local JiNao monitors. Remote Management Application monitors could communicate peer-to-peer with other Remote Management Application monitors via the standard ManagerToManager portion of the SNMP protocol to aggregate results to support regional detection. Furthermore, Remote Management Application monitors could communicate hierarchically to develop a more global view of potential misuse in the network. Although the *JiNao Report* states that a hierarchical system had not yet been created, the report is clear that the architecture described was suitable for implementing such a hierarchy. Furthermore, because the *JiNao Report* explicitly directed one of skill to use the known hierarchical architecture of the SNMP framework, one of skill in the art would have understood this reference to disclose and enable hierarchical JiNao monitors.

32. JiNao processed network packets. The messaging infrastructure for internal JiNao modules used a generic Protocol Data Unit (PDU), so they could be tailored for any protocols. MCNC/NCSU proposed to deliver solutions for the OSPF and SNMP protocols, and the *JiNao Report* mentions the possibility of using JiNao to process HTTP packets to protect web servers. In addition, the *JiNao Report* explicitly states that JiNao monitored data volume.[8]

33. JiNao analyzed network traffic through a combination of anomaly detection based on the NIDES statistical algorithms and misuse detection. The Simple Misuse Detection module (called "Prevention Module" in the report) used simple rules that could quickly evaluate individual packets for clear security violations. The Stateful Misuse Detection Module (called "Protocol Analysis" in the report) used a collection of Finite State Machines (FSMs) to detect more complex attacks that can span multiple packets. The anomaly detector was based on the NIDES statistical algorithms.

34. The Protocol Analysis module was a collection of finite state machines (FSM). Each FSM was a signature designed to detect a specific attack. UC Davis had also done research on methods to detect attacks through protocol analysis, and thus I am familiar with this type of analysis. Typically, such analysis sets forth a set of rules describing the "correct" behavior of a particular protocol, and then looks for deviations from this known proper behavior. Such analysis may also look for known exploitations of known vulnerabilities of a particular protocol.

35. The Local Decision Module correlated the results from the anomaly and misuse detectors. Hierarchically higher monitors would use similar detection/analysis functions to "correlate intrusion events among several routers."[9] For example, as described in the *JiNao Report*, JiNao could correlate conflicting information from

[8] *JiNao Report* at 19.
[9] *JiNao Report* at 13.

multiple JiNao systems to determine that a router between those two systems (Bob) was bad.[10]

36. JiNao could respond to suspicious events in several ways. First, the Local Decision Module could update the MIB Database, thus making the information available to hierarchically higher monitors. Second, the Local Decision Module could immediately automatically alert hierarchically higher monitors with an SNMP trap message. Third, the Local Decision Module could direct the local analysis modules to modify their analyses. Fourth, the Local Decision Module could raise an alert to the security officer via an email or a graphical user interface. Fifth, the Local Decision Module could interact directly with the router to modify its settings. Sixth, the Simple Misuse Detection module (called "Prevention Module" in the document) could direct the Interception/Reception module not to forward the packet to the router's software.

37. The JiNao modules would satisfy the "monitor" limitation under any of the parties' definitions of that term.

### B. EMERALD 1997

38. This section covers the EMERALD design as described in the paper titled "EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances," published in the Proceedings of the 20th National Information Systems Security Conference in October of 1997 ("*Emerald 1997*"). Phil Porras, one of the named inventors of the patents-in-suit, and Peter Neumann are the authors of this paper. The primary sponsor for this work was the Department of Defense via the Defense Advanced Research Projects Agency (DARPA).

39. *Emerald 1997* describes an architecture collectively called EMERALD for monitoring an enterprise network. EMERALD analyzed an input stream such as network

[10] *JiNao Report* at 35.

packets, and applied anomaly detection, misuse detection, correlation, and aggregation to detect possible misuse. EMERALD's internal modules used a well-defined API to pass messages between modules; EMERALD also used a well-defined protocol to support hierarchical and peer-to-peer communications to provide scalability and detect larger scale attacks.

40. EMERALD as described in this publication had the following salient features:

- Monitored network services and network components such as routers and gateways.
- Analyzed network packets, audit logs, application logs, and results of other intrusion detection sensors; these data sources are referred to as an event stream.
- Used anomaly detection based on the NIDES algorithms to analyze the event stream.
- Used signature detection to analyze the event stream for known examples of misuse.
- Provided an Application Programming Interface (API) to allow third-party modules and sensors to participate in an EMERALD system.
- Correlated analysis from the anomaly detection engine, the signature detection engine, and potentially other analysis engines.
- Supported hierarchical monitoring beginning at the lowest level (service monitor), with the next higher level being a domain monitor, and the highest level being an enterprise monitor.

41. The system described in *Emerald 1997* shared the basic elements of SRI's previous work in IDES and NIDES, but it was designed to be distributed and independent of any particular data source. The paper states that EMERALD's distributed design addresses scalability. Designing the system to be independent of the underlying data source makes it easier to incorporate other data sources, including results from other intrusion detection sensors.

42. EMERALD took the basic NIDES architecture – an anomaly engine, a signature/misuse engine, and a resolver – and generalized it for any input stream. The earlier SRI intrusion detection systems (IDES, NIDES and Safeguard) shared a similar architecture, but they were largely focused on an input stream of host-generated data (e.g., audit trails) to analyze host-related subjects (e.g., user accounts and processes). EMERALD generalized the approach so that the system was not tied to any specific input stream or subject matter.[11]

43. Furthermore, because the input stream was generalized, the output event stream for one monitor could be used as the input stream for another monitor. Because EMERALD monitors could be composed in this manner, a hierarchy of monitors could be created, and this hierarchical approach in turn created the ability to scale the analysis to larger environment.

44. An EMERALD "Service Monitor" read input streams such as network packets, and then analyzed that input stream with an anomaly detection engine and a signature detection engine. The results of these analyses engines were passed to the Resolver, which could correlate the results, take some type of response, or pass its analysis to hierarchically higher monitors.

45. The next level of monitor in the hierarchy was called the "Domain Monitor", and it analyzed and correlated the results from the lower level Service Monitors. The Domain Monitor shared the same architecture as the Service Monitor, including an anomaly detection engine, a signature/misuse detection engine, and a Resolver. The Domain Monitor could also pass the results of its analysis to other Domain Monitors via peer-to-peer sharing, or it could pass the results to a hierarchically higher Enterprise Monitor.

[11] *Emerald 1997* at Fig. 1 shows the "generic EMERALD monitor architecture."

46. The top level of monitor in the hierarchy was called the "Enterprise Monitor". The Enterprise Monitor analyzed and correlated the Domain Monitors' results. The Enterprise Monitor shared the same architecture as the Domain and Service Monitors, namely, an anomaly detection engine, a signature/misuse detection engine, and a resolver.

47. EMERALD processed any type of input stream. Examples in the *Emerald 1997* paper included audit data, network datagrams (i.e., packets), SNMP traffic, application logs, and results from other intrusion detection sensors.[12]

48. EMERALD used a combination of anomaly detection, signature detection, and a resolver to analyze the input streams. The signature/misuse engine detected known intrusive behavior. The anomaly engine detected unusual activity that might indicate previously unknown intrusive behavior. The resolver integrated and correlated information from the anomaly and signature engines and potentially from other monitors as well.

49. The signature analysis engine is described as mapping an incoming event stream against "abstract representations of event sequences that are known to indicate undesirable activity."[13] The objectives of the signature analysis depend on which level in the hierarchy the analysis is operating at: service monitors target known attacks on network services and infrastructure, whereas above the service layer, the signature engines scan aggregated reports for more global coordinated attack scenarios.

50. The anomaly detection engines are described as using the statistical profiling type of anomaly detection developed in NIDES. The paper notes that the "underlying mechanisms" of NIDES are "well suited to the problem of network anomaly

[12] *Emerald 1997* at 356.
[13] *Emerald 1997* at 359.

detection, with some adaptation."[14] *Emerald 1997* then described the modifications to the statistical profiling mechanism needed in order to achieve the generality of input desired in EMERALD.

51. EMERALD's resolver was responsible for responding to suspicious behavior. The paper describes several different actions the resolver could take in response to detected suspicious activity including: (1) closing connections, (2) terminating processes, (3) calling a host's integrity checker to verify operating state, (4) propagating information to other monitors, (5) modifying the analysis of its detection engines, and (6) sending information to the user interface.[15]

52. Furthermore, given the similarity in language between *Emerald 1997* and the patents, my opinions on invalidity and obvious with regard to *Emerald 1997* and all other NIDEs and Emerald –related references do not change regardless of which claim construction position is ultimately adopted by the Court.

53. SRI has claimed that *Emerald 1997* does not teach the specific categories of network traffic data called out in, for example, '338 claim 1, '203 claim 1, and '615 claim 1.[16] First, it is important to point out that monitoring these particular types of "measures" of network packets or "network traffic data" was not new or novel – all of them had been monitored by other systems before. The inventors have acknowledged this fact for many of the claimed categories.[17]

54. Second, as explained in the Expert Report of Frederick Avolio, *Emerald 1997* explicitly stated that the disclosed EMERALD system could monitor and analyze "network infrastructure" including firewalls. *Emerald 1997* also disclosed monitoring an

[14] *Emerald 1997* at 359.
[15] *Emerald 1997* at 361.
[16] *See* SRI International, Inc.'s "Amended" Response to Symantec's Invalidity and Inequitable Conduct Contentions.
[17] Porras Tr. 289-295, 444-454; Valdes Tr. 283-287.

event stream from an application log, which would include a firewall log. Firewalls in 1997 provided a common set of monitoring and logging features which were well-known in the art at the time. Thus, one of skill would have understood from the *Emerald 1997* disclosure that one should monitor network connections, including network connection requests and denials, and data transfers, including network packet data volume/network packet data transfer volume. I have reviewed Mr. Avolio's report and spoken with him in detail about it. I agree with and adopt in my own report his report, including his analysis and conclusions. I also adopt the references he relied upon in reaching his conclusions.

55. In addition, *Emerald 1997* disclosed monitoring an event stream of SNMP traffic[18] and monitoring network infrastructure.[19] The Internet Standards (RFCs) for SNMP traffic and related network infrastructure management data provide monitoring for a variety of different categories of network traffic data. The claimed categories of network traffic data would have been understood by one of ordinary skill in the art to be disclosed by this explicit reference to monitoring SNMP traffic and network infrastructure.

56. Furthermore, the types of network traffic one should monitor to detect suspicious activity or network intrusions flow naturally from the types of attacks that one is looking for. As the intrusion detection and computer security fields developed, practitioners in the field began cataloging and tracking security-related intrusions seen by different computer networks. For example, after the Morris worm incident in November 1988, which caused extensive damage to different internet systems, DARPA called on the Software Engineering Institute at CMU to set up a center to coordinate communication among experts during security emergencies.[20] Known as the Computer Emergency

[18] *Emerald 1997* at 356.
[19] *Emerald 1997* at 355.
[20] http://www.cert.org/meet_cert/meetcertcc.html

Readiness Team ("CERT"), CERT issued public advisories to warn of new attacks or vulnerabilities.

57. For example, in September 1996, CERT first issued a warning about TCP SYN flooding and IP Spoofing attacks.[21] This advisory explained the SYN flooding attack, in which numerous requests to open a TCP connection are sent that cannot be responded to, flooding the system with half-open connections and making it difficult for the system to continue to communicate. One of skill in the art at the time would have known about SYN flooding attacks and understood that one should monitor network connection requests and denials to detect such an attack. A variety of other types of network attacks were also well-known at the time, which would have similarly indicated different categories of network traffic to monitor.[22]

58. To the extent that the various categories of network traffic data or measures of network packets are not disclosed in *Emerald 1997*, it would have been obvious to one of skill in the art to combine *Emerald 1997* with any of the many well-known firewalls, intrusion detection research systems, commercial intrusion detection systems, and IETF defined standards on what to monitor for network infrastructure, all of which were already monitoring these network traffic data categories. In fact, *Emerald 1997* explicitly pointed the reader to other systems, including NSM.[23] For example, both ISS RealSecure and NetRanger were well-known network intrusion detection systems, and one of skill upon reading *Emerald 1997* would have been motivated to look at the traffic data being monitored by such systems in order to select the best categories of

[21] CERT Advisory CA-1996-21 [SYM_P_0548726-734]. *See also* Schuba et al., "Analysis of a Denial of Service Attack on TCP," Proc. of the 1997 IEEE Symposium on Security and Privacy, Oakland, CA, 208-23 (May 4-7 1997) [SYM_P_0535408-28].
[22] Numerous other attacks were well-known at the time. *See, e.g.*, CERT Advisory CA-1996-26 "Denial-of-Service Attack via ping, Dec. 18, 1996 [SYM_P_0548718-725] (discussing ICMP error packets and oversized ICMP datagrams).
[23] *Emerald 1997* at 364.

traffic for use with EMERALD. Indeed, I understand that the inventors in 1997 were actually working with and looking at Sun Microsystem's SunScreen firewall, further supporting my opinion that firewalls and other IDS systems would have been obvious sources of information about useful network traffic data categories.[24]

59. I also understand that SRI claims that the *Emerald 1997* paper does not provide an "enabling disclosure" of a statistical detection method.[25] I disagree. *Emerald 1997* included an entire section entitled "Scalable Profile-Based Anomaly Detection" which describes how EMERALD will incorporate and modify the well-known NIDES algorithms for statistical profiling. *Emerald 1997* also pointed the reader to additional references on the NIDES algorithms. Furthermore, other additional SRI publications had also already disclosed the NIDES algorithms in detail.[26] To the extent that the specific algorithms for statistical profiling are not incorporated-by-reference into *Emerald 1997*, it would have been obvious to one of skill in the art to combine *Emerald 1997* with prior NIDES-related publications, since *Emerald 1997* states that the NIDES algorithms form the basis for the statistical profiling disclosed. One of skill would have been motivated to further investigate these specific algorithms in order to improve the performance of the system.

[24] Valdes Tr. 68-69; 12-123. *See also Emerald – Conceptual Design 1997* at p. 88. [SRI 012400].

[25] *See* SRI International, Inc.'s "Amended" Response to Symantec's Invalidity and Inequitable Conduct Contentions.

[26] *See, e.g.*, A. Valdes and D. Anderson, *Statistical Methods for Computer Usage Anomaly Detection Using NIDES*, Proc. of the Third International Workshop on Rough Sets and Soft Computing, January 1995 ("*Statistical Methods*") [SYM_P_0068937-942]. Mr. Valdes testified that the equations disclosed in this paper were suitable for network monitoring. Valdes Tr. 344-346. Furthermore, the patents' specification explicitly states that the techniques described in this paper may be used for the "profile engine" which "can profile network activity via one or more variables called measures." '338 col. 5:43-50.

### 1. Emerald 1997, Intrusive Activity 1991, NIDES 1994

60. *Emerald 1997* also references two additional publications in both the text of the paper itself, and in the list of references. *Emerald 1997* explains that the statistical algorithms in H. Javitz and A. Valdes, "The NIDES statistical component description and justification," Technical report, Computer Science Laboratory, SRI International, Menlo Park, CA March 1994 (*"NIDES 1994")* provides the foundation for the profile-based anomaly detection in *Emerald 1997*:

> "Requirements for an anomaly-detection system that became IDES were documented in [6]. This research led to the development of the NIDES statistical profile-based anomaly-detection subsystem (NIDES/Stats), which employed a wide range of multivariate statistical measures to profile the behavior of individual users [9]."[27]

61. In addition, *Emerald 1997* also directs one to the Network Security Monitor (NSM) system for analysis of network traffic, and identifies L.T. Heberlein, B. Mukherjee, K.N. Levitt., "A Method to Detect Intrusive Activity in a Networked Environment," Proc. 14th National Computer Security Conference, pp. 362-371, Oct. 1991 ("*Intrusive Activity 1991*") as describing NSM:

> "[T]he Network Security Monitor [7] seeks to analyze packet data rather than conventional audit trails…"[28]

62. Given these explicit references in *Emerald 1997* to both *NIDES 1994* and *Intrusive Activity 1991*, these three references should be considered to be a single disclosure for purposes of anticipation. In the alternative, the citations above provide a motivation to combine these three references in order to make and improve the statistical profiles and network traffic monitoring claimed in the patents-in-suit.

[27] *See Emerald 1997* at 359.
[28] *See Emerald 1997* at 364.

## V. OBVIOUSNESS

63. In the previous sections, I have already explained many reasons why one of skill in the art would have been motivated to combine a particular reference or system with another. In addition to those explanations, I will further elaborate some factors that relate to the obviousness of the alleged inventions. In general when addressing obviousness, I have considered the issue of what a person of ordinary skill in the art prior to November 9, 1997 would have done if confronted by the problem of detecting network intrusions if that person had no knowledge of the patents-in-suit. I have further considered the problem of detecting network-related intrusions across an enterprise network, spanning multiple domains, and how that problem would have been confronted by one of skill in the art in the relevant timeframe.

64. It is important to point out that by November 1997, there was a very rich body of prior art in the network monitoring and intrusion detection fields. Many different individuals and research groups had been working on these issues for many years. Thus, one of ordinary skill in the art would have known about many such systems and past work. Given this large body of prior work, it is impossible for me to enumerate all the possible, likely combinations. My report discusses merely a sample of the possibilities.

65. As discussed earlier with regard to *Emerald 1997*, much of the motivation to look at particular categories of network traffic stems directly from the types of attacks that can be launched against networks. As I understand the alleged inventions of the patents-in-suit, a key issue is whether one of skill in the art would have been motivated to use the claimed network traffic data categories as an input into a network-based intrusion detection system. In my opinion, the claimed network traffic data categories broadly direct one to look at parts of mainly network packet headers, or to look at the volume of data in a packet. If looking for network-based intrusions, or indeed any type of activity

on a network, one of skill in the art prior to November 1997 would naturally have looked at these parts of a network packet.

66. A network packet typically consists of a header portion and a data portion. The structure of network packets was well-known and indeed trivial in the computer science field in the 1990s. In addition, monitoring and parsing different portions of network packets to gain relevant information about a particular part of a network packet was similarly common in the computer science field.

67. By 1997, it was well-known that attackers could use either the header portion of a packet or a data portion of a packet to carry out an attack.[29] While attacks like SYN floods, IP scans, port scans, and SATAN scans primarily relied on the header portion of network packets, other attacks such as buffer overflow (also known as buffer-overrun) exploits, password cracking, and worms relied heavily on transporting data or code within the data portion of network packets.

68. Attacks that relied primarily on the header portion of network packets used the various data fields contained in the header to carry out the attack. These fields are ordinarily used by various network protocols to administer the transfer of data via network packets. For example, some header fields—such as the SYN and ACK fields in a TCP header—relate to administering the establishment of network connections. Other header fields—such as the source IP address and destination IP address fields in an IP header—relate to administering the transfer of data from one computer to another over a network. Because well-known attacks used such fields in malicious ways, one of ordinary skill in the art would have been motivated to monitor header fields using both profile-based anomaly detection and signature detection.

[29] NetRanger 1.3.1 User's Guide at 4-61 (see, e.g., IP Fragment attack and FTP CWD ~root exploit).

69. Other attacks relied primarily on the data portion of network packets to transfer malicious data or code. As noted previously, it was well-known by the early 1990s that several network-based attacks transported malicious code, data, or commands via the data portion of network packets. Because these attacks placed malicious code, data, or commands in the data portion of network packets, one of ordinary skill in the art would have been motivated to monitor the data portion of network packets using both profile-based anomaly detection and signature detection.

70. As I described earlier, network protocols are typically organized as a layered stack, with each protocol being built on the protocol or protocols directly beneath it. It was well-known by 1997 that attacks could be carried out at any layer of the network protocol stack. For example, while SYN attacks were carried out using transport layer TCP flags relating to connection establishment, many attacks directed at FTP servers were carried out using application layer FTP-specific commands. Because each and every network protocol layer was the subject of attacks, one of ordinary skill in the art would have been motivated to monitor each and every layer using both statistical profile-based anomaly detection and signature detection.[30]

71. To summarize, in order to increase the number of attacks that could be detected, one of ordinary skill in the art would have been motivated to use intrusion detection systems to analyze all portions of a network packet, including the header and data portions of packet formats from each network protocol layer. The nature of the problem to be solved would have led one of skill to examine the claimed network traffic data categories.

[30] The need to monitor each and every network protocol layer is related to the need to monitor both the header and data portions of network packets, since packets at a given layer typically encapsulate packets at a higher layer. For example, in order to monitor transport-layer TCP segments, one has to extract those segments from the data portion of network-layer IP packets.

72. By 1997, it was also well-known to those of skill in the art that attackers targeted all types of network entities, including specialized entities like routers, gateways, proxy servers, firewalls, proxy servers, and any other device connected to a network. There were several reasons why attackers targeted a wide variety of network entities. First, because of the prevalence of use of standard network protocols, an attack that exploited one of these standard protocols could typically be launched against the wide array of network entities using that protocol. For example, a SYN flood that was initially developed to attack servers utilizing TCP could just as easily be launched against a handheld networked device that also utilized TCP. Thus, just as network-based intrusion detection leveraged the network protocols to monitor several different types of network entities, attackers could also leverage the use of these protocols to expand their base of attack targets.

73. Second, attackers had an incentive to attack these specialized network entities that comprised the internet infrastructure itself. Infrastructure entities like routers and gateways were attractive targets because an attack on those targets would be felt by every network entity that used the router or gateway as an intermediary for communication to the rest of the network. In other words, attacks on routers and gateways potentially would be more widely felt. Because of the importance of these entities that comprised the network infrastructure itself, one of ordinary skill in the art would have been motivated to protect these entities using intrusion detection systems.

74. By November 1997, there also existed a strong motivation to design intrusion detection systems to interoperate with both other intrusion detection systems and other types of network security devices. Standards-based efforts such as CIDF had emerged to promote interoperability through the use of common protocols and common APIs. There were several factors driving this move towards greater interoperability. From a technical standpoint, greater interoperability enhanced the ability of an intrusion

detection system to analyze information from a wider range of network security devices (like firewalls). Greater interoperability also enhanced scalability by allowing intrusion detection systems to work and communicate with each other regardless of whether these systems were administered by the same party. In this way, cooperation enabled by greater interoperability would allow intrusion detection systems to scale up to larger networks. From a commercial standpoint, greater interoperability allowed new products to work with and leverage existing deployments of heterogeneous network security solutions.

75. Both the SRI IDES/NIDES/EMERALD team and UC Davis' Computer Science Laboratory were well-known players in the intrusion detection space in the 1990s. Both groups published, presented publicly at conferences, and were involved in related DARPA projects. One of ordinary skill in the art would have known about these groups, and would have considered them to be doing related work. Given that both teams were attempting to solve problems relating to network intrusion detection, it would have been obvious to one of skill in the art that it would be worthwhile to combine systems from these two groups.

76. In addition, given the number of "surveys" and compilations discussing various different intrusion detection systems in existence prior to November 9, 1997, there was ample direction and motivation to combine such systems. One such example[31] is a 1994 paper on which I am a named author: B. Mukherjee, L. Todd Heberlein and K. N. Levitt, "Network Intrusion Detection," IEEE Network May/June 1994 ("*NID 1994*").[32] This paper notes that "[t]he intrusion detection problem is becoming a

[31] For additional examples in 1997, see L. Todd Heberlein, Network Radar presentation, 24 July 1997; and http://web.archive.org/web/19971011083618/www.hokie.bs1.prc.com/ia/N2-TODD.htm

[32] I understand that a copy of this publication was actually produced from the files of Mr. Porras, one of the named inventors. *See* SRI 058251.

challenging task due to the proliferation of heterogeneous computer networks."[33] The paper also points out that it is common to combine statistical and rule-based systems, and provides the reader with several examples of each: "[t]ypically, IDSs employ statistical anomaly and rule-based misuse models in order to detect intrusions…" "[i]n this paper, several host-based and network-based IDSs are surveyed, and the characteristics of the corresponding systems are identified."[34] The paper encourages further investigation: "new and more-effective detection strategies must be investigated."[35] The paper specifically encourages investigation into intrusion detection for large networks: "much more research is expected to be conducted, e.g., how can the intrusion-detection concept be extended to arbitrarily large networks…"[36]

77. Given these specific pointers in *NID 1994* to a variety of existing intrusion detection systems, as well as the specific direction to combine statistical and rule-based approaches and expand IDS to better cover arbitrarily large networks, one of skill in the art would have been motivated to combine the systems discussed in this paper, as well as other related systems, to achieve the goals of the paper. *NID 1994* encouraged one of skill to investigate and combine existing systems.

### A. Secondary Considerations

78. I have been informed that it is appropriate in analyzing the obviousness of an alleged invention to consider "secondary considerations" of non-obviousness. I understand that secondary considerations include: commercial success of the invention; satisfying a long-felt need; failure of others to find a solution to the problem; and copying of the invention by others. Other secondary considerations include licensing by competitors and contemporaneous recognition of the inventor's achievements.

[33] *NID 1994* at 26.
[34] *NID 1994* at 26.
[35] *NID 1994* at 41.
[36] *NID 1994* at 41.

79. Based upon my review and understanding of the facts, as well as my own personal knowledge, I am not aware of any secondary considerations supporting a finding that the patents-in-suit were not obvious.

80. In my opinion, the Emerald system disclosed in the patents-in-suit did not satisfy any long-felt need that I was aware of. I was very familiar with intrusion detection systems in the late 1990s and was deeply involved in the field. I do not recall anyone identifying any "need" fulfilled only by the Emerald system. My peers in the intrusion detection field were aware of the Emerald system through conferences, but I do not recall any particularly unique feature of Emerald that was considered extremely valuable to the intrusion detection community.

81. Furthermore, I am not aware of anyone copying the Emerald system.

82. Many other systems, some of which have been discussed in this report, were able to successfully detect computer intrusions and in particular network attacks. For example, the NSM system and its many different incarnations, such as ASIM, were successful in detecting network attacks. (NSM became ASIM, which as indicated in Mr. Teal's report was widely used by the Air Force). Thus the government, which funded Emerald, was actually using other intrusion detection systems, including some that started out as research-oriented projects such as NSM. In addition, DIDS, another research-oriented system from UC Davis, was very successful in developing useful correlation features to track users across different systems, solving a key problem in computer security.

83. It is important to recognize that there is a difference between actual commercial products and research funded by the government. Laudatory statements regarding research projects do not necessarily translate into success in the commercial world, because research projects are evaluated on a different set of metrics than actual commercial projects. Research systems typically are run in a limited, lab environment,

whereas commercial systems need to run robustly in a messy, real-world environment. Furthermore, research projects typically only address a small part of the problem, whereas commercial systems need to have a rich supporting infrastructure to make them usable products.

84. In the commercial realm, the NetRanger system was a commercial success used by Fortune 500 companies to protect their networks from intrusions. NetRanger was also successful in government testing, with the DOD/SPOCK report stating:

> "Results of the tests clearly demonstrated that when properly configured, the NetRanger hardware/software package:
>
> 1) Can be used to detect, report, and act on intrusion related activities launched across a network with a high degree of accuracy,
>
> 2) Would detect all attempted penetrations signatures contained in the default list as installed in the NetRanger for this demonstration,
>
> 3) Can be used to provide practical and effective intrusion detection, reporting, and selected automatic response actions."[37]

In addition, the DOD/SPOCK report concluded "[i]n the true sense, this suite of tests proved the viability of Real-Time Network Intrusion Detection and Response for implementation today, in a warfighter networked environment."[38]

85. I do not believe that the EMERALD system ever achieved the success that NSM/ASIM and NetRanger did.

## VI. ENABLEMENT AND SUFFICIENT WRITTEN DESCRIPTION OF THE PATENTS-IN-SUIT AND PRIOR ART

### A. LEGAL STANDARD

86. I understand that the specification of a patent must provide an enabling disclosure. I understand that this requires that a person of skill in the art, using

[37] DOD/SPOCK Report at 2 [SYM_P_0074255- SYM_P_0074481 at SYM_P_0074263].
[38] DOD/SPOCK Report at 5.4 [SYM_P_0074255- SYM_P_0074481 at SYM_P_0074287].

knowledge available to them and the disclosure in the patent, could make and use the invention without undue experimentation. I also understand that the enablement standard for prior art publications is similar to that required for a patent specification.

87. I have been informed that the factors to be assessed in determining whether experimentation is "undue" include: the quantity of experimentation necessary, the amount of direction or guidance presented, the presence or absence of working examples, the nature of the invention, the state of the prior art, the relative skill of those in the art, the predictability or unpredictability of the art, and the breadth of the claims.

88. I also understand that the specification of a patent must describe the subject matter claimed in the patent in a manner that conveys to one of skill in the art that the inventors had possession of the subject matter claimed at the time the patent application was filed.

### B. ANALYSIS OF ENABLEMENT / WRITTEN DESCRIPTION OF *EMERALD 1997* AND *LIVE TRAFFIC ANALYSIS*

89. I understand that SRI contends that the disclosures in certain prior art references, including *Emerald 1997*, are not enabling for certain claim limitations.[39] It is my opinion that the prior art references discussed in my report, including *Emerald 1997* and *Live Traffic Analysis*[40], are enabling and provide a disclosure at the same general level of detail as found in the specification of the patents-in-suit. In particular, given the overall similarity between the disclosures in *Emerald 1997* and the patents-in-suit, including substantial portions of identical text and identical figures, it is not plausible to

[39] For example, SRI has claimed that *Emerald 1997* does not provide an enabling disclosure of a statistical detection method. *See* SRI International, Inc,'s "Amended" Response to Symantec's Invalidity and Inequitable Conduct Contentions (Dec. 16, 2005).

[40] P. Porras and A. Valdes, "Live Traffic Analysis of TCP/IP Gateways," http://www.sdl.sri.com/projects/emerald/live-traffic.html, Internet Society's Networks and Distributed Systems Security Symposium, Nov. 10, 1997 ("*Live Traffic Analysis*").

provides detailed diagrams of the flow of information through the system (*see, e.g.*, Fig. 1 at page 4 and Fig. 3 at page 17). The report also provides a detailed description of the statistical analysis module used, including the algorithms used (*see* pages 18-24). Based upon my review of this document, it is my opinion that the *JiNao Report* is enabling and provides a disclosure at a similar level of detail to the specification of the patents-in-suit.

95. To the extent that SRI claims that the *JiNao Report* is not enabling, my opinion is that this would necessitate a finding that the patents-in-suit themselves similarly do not satisfy the enablement and written description requirements.

## VII. INVENTORS' FAILURE TO DISCLOSE BEST MODE

### A. Legal Standard

96. I understand that the patent laws require that if an inventor knows of a best mode of practicing the claimed invention, the inventor must disclose that best mode. I also understand that the legal standard for this best mode requirement involves two factual inquires: (1) a subjective determination of whether the inventor had a best mode of practicing the claimed invention; and (2) if the inventor had a best mode of practicing the claimed invention, an objective determination of whether the best mode was disclosed in sufficient detail to allow one skilled in the art to practice it. I have been asked to render an opinion regarding both of these inquires.

### B. Documents Relied Upon and Methodology

97. As part of my inquiry into whether the inventors satisfied the best mode requirement, I have reviewed many documents. For the purposes of this litigation, SRI has placed into escrow a computer containing source code[46] related to the patents-in-suit.

[46] In this report, I am using the term "source code" to encompass not only the text files that are compiled or interpreted in order to generate an executable software component, but also any associated initialization files, configuration files, or other types of input files that are read in by the software component.